

Avoiding the redundancy of adding the object files to both the primary project and its unit test

 devblogs.microsoft.com/oldnewthing/20230222-00

February 22, 2023



Raymond Chen

A customer was hoping for a simpler way to structure their unit tests: They had a Visual Studio solution with two projects, Contoso and ContosoUnitTest. Every time they add a new cpp file to Contoso, they have to add a reference to that same cpp file to the ContosoUnitTest.

Is there an easier way to do this? Maybe a way to set a wildcard in the ContosoUnitTest project so it just slurps up all of the cpp files in the Contoso directory?

(This problem and its solution applies to any build system, but I chose Visual Studio for concreteness.)

I don't think there's a wildcard option in Visual Studio,¹ but even if it did, I don't feel comfortable using it, because it means that any random file not part of the project that happens to be created in the directory will get scooped up into the unit test project.

As with many problems in computer science, this can be solved by adding another level of indirection. In this case, what you do is create a new project called ContosoCore or ContosoLib or something like that. This project includes all the cpp files and produces a library file. You then consume that library from the Contoso and ContosoUnitTest projects.

When you add a new cpp file to your project, add it to the ContosoLib project, and that will add it to both Contoso and ContosoUnitTest.

This also has the benefit that the cpp files are compiled only once (by ContosoLib), and the compiled cpp files are then consumed by the production Contoso project and the ContosoUnitTest.

¹ Current documentation says that Visual Studio IDE doesn't support wildcards.